LAMPIRAN

home.dart

```
import
'package:easy_localization/easy_loc
alization.dart';
import
'package:flutter/material.dart';
import
'package:flutter_toggle_tab/flutter_t
oggle_tab.dart';
import
'package:infografis_daging_sapi/sty
les/constant.dart';
import
'package:infografis_daging_sapi/wi
dgets/headerHome.dart';
import
'package:infografis_daging_sapi/wi
dgets/menuHome.dart';

class HomeScreen extends
StatelessWidget {
 const HomeScreen({Key? key}) :
super(key: key);

 @override
 Widget build(BuildContext
context) {
   return Scaffold(
     body: Column(
       children: <Widget>[
         Flexible(
           child: Stack(children:
<Widget>[
             Positioned.fill(
               top: 210,
               child: Image(
                 image:
AssetImage('assets/images/backgro
und.jpg'),
                 fit: BoxFit.fill,
               ),
             ),
             Column(
               children: <Widget>[
                 HeaderHome(),
                 SizedBox(height: 20),
                 MenuWidget(
                   text:
"menu.infografis".tr(),
                   url: "/infografis",
                   img:
"assets/images/cow-menu.png",
                 ),
                 SizedBox(height: 20),
                 MenuWidget(
                   text:  tr("menu",
gender: "m&f"),
                   url: "/mitosFakta",
                   img:
"assets/images/mitos-menu.png",
                 ),
                 SizedBox(height: 20),
                 MenuWidget(
                   text:  tr("menu",
gender: "olahan"),
                   url: "/anekaOlahan",
                   img:
"assets/images/olahan-menu.png",
                 ),
                 SizedBox(height: 20),
                 MenuWidget(
                   text:  tr("menu",
gender: "tentang"),
                   url: "/tentang",
                   img:
"assets/images/tentang-menu.png",
                 ),
                 Expanded(child:
SizedBox(height: 20)),
                 FlutterToggleTab(
                   width: 50,
                   borderRadius: 20,
                   initialIndex:
context.locale.toString() == "id" ? 0
: 1,

selectedBackgroundColors:
[primaryColor],

unSelectedBackgroundColors:
[bgColor],
                   selectedTextStyle:
TextStyle(
                     color: Colors.white,
                     fontSize: 18,
                     fontWeight:
FontWeight.w500),
                   unSelectedTextStyle:
TextStyle(
                     color: primaryColor,
                     fontSize: 14,
```

```
                    fontWeight:
FontWeight.w400),
                labels:     ["Indonesia",
"English"],
                selectedLabelIndex:
(index) {
                  context
                    .setLocale(index
== 1 ? Locale('en') : Locale('id'));
                },
              ),
              SizedBox(height: 20),
            ],
          )
        ]),
      ),
    ],
  ),
 );
 }
}
```

infografis.dart

```
import
'package:flutter/material.dart';
import
'package:easy_localization/easy_loc
alization.dart';
import
'package:flutter_bloc/flutter_bloc.d
art';
import
'package:infografis_daging_sapi/bl
oc/infografis_bloc.dart';
import
'package:infografis_daging_sapi/bl
oc/resep_bloc.dart';
import
'package:infografis_daging_sapi/m
odels/daging_models.dart';
import
'package:infografis_daging_sapi/m
odels/infografis_models.dart';
import
'package:infografis_daging_sapi/sty
les/constant.dart';
import
'package:infografis_daging_sapi/m
odels/sapiCoordinate.dart';
import
'package:infografis_daging_sapi/src
/imageMap.dart';
import
'package:infografis_daging_sapi/wi
dgets/alertError.dart';
import
'package:infografis_daging_sapi/wi
dgets/deskripsiInfografis.dart';
import
'package:infografis_daging_sapi/wi
dgets/listDagingInfografis.dart';
import
'package:infografis_daging_sapi/wi
dgets/listResepInfografis.dart';
import
'package:infografis_daging_sapi/wi
dgets/loading.dart';

class InfografisScreen extends
StatelessWidget {
 const InfografisScreen({Key?
key}) : super(key: key);

 @override
 Widget build(BuildContext
context) {
  List<InfografisModel>
infografisList =
    context.select<InfografisBloc,
List<InfografisModel>>(
      (infografisBloc) =>
infografisBloc.state.infografisList);
  List<DagingModel> dagingList
=
    context.select<InfografisBloc,
List<DagingModel>>(
      (infografisBloc) =>
infografisBloc.state.dagingList);

  bool isDataLoaded =
context.select<InfografisBloc,
bool>(
    (infografisBloc) =>
infografisBloc.state.isDataLoaded);

  int selectedList =
context.select<InfografisBloc, int>(
    (infografisBloc) =>
infografisBloc.state.selectedList);

  if (!isDataLoaded &&
infografisList.length <= 0)

context.read<InfografisBloc>().add
(InfografisLoaded());

  final List<Color> colors =
List.generate(
   points.length,
```

```
    (index) => index == selectedList
        ? Color.fromRGBO(255, 255, 255, 0.8)
        : Colors.transparent,
   );

   final List<Path> polygonRegions = points.map((e) {
     Path p = Path();
     p.addPolygon(e, true);
     return p;
   }).toList();

   return Scaffold(
     appBar: AppBar(
       title: Text(
         "menu.infografis",
         style:  TextStyle(fontWeight: FontWeight.bold),
       ).tr(),
     ),
     backgroundColor: bgColor,
     body: BlocListener<InfografisBloc, InfografisState>(
       listener: (context, state) {
         print(state.error);
         if (state.isLoading && !state.isDataLoaded) {

Navigator.of(context).push(Loading());
         } else if (!state.error.status && !state.isDataLoaded) {
           Navigator.pop(context);
           alertError(context);
         } else if (!state.isLoading &&
             state.isDataLoaded &&

!ModalRoute.of(context)!.isCurrent) {
           Navigator.pop(context);
         }
       },
       child: Column(
         children: [
           SizedBox(height: 20),
           Card(
             margin: const EdgeInsets.symmetric(horizontal: 16),
             shape: RoundedRectangleBorder(
               borderRadius: BorderRadius.circular(20.0),
             ),
             child: Column(
               children: [
                 ListTile(
                   leading: Icon(
                     Icons.info,
                     color: Colors.white,
                   ),
                   title: Text(
                     "info_klik",
                     style: TextStyle(color: Colors.white, fontSize: 14),
                   ).tr(),
                   tileColor: primaryColor,
                 ),
                 Padding(
                   padding: const EdgeInsets.symmetric(horizontal: 16),
                   child: ImageMap(
                     imagePath:

"assets/images/cow-${context.locale.toString()}.png",
                     imageSize: Size(1000, 996),
                     onTap: (i) {

print(infografisList);
                       if (infografisList.length > 0) {
                         context

.read<InfografisBloc>()

.add(SelectedChange(i));
                         context

.read<InfografisBloc>()

.add(FilterDagingList(dagingList, i + 1));

_modalBottomSheetInfografis(
                           context,
                           () => context

.read<InfografisBloc>()

.add(SelectedChange(-1)));
                       } else {
```

```
alertError(context);
                    }
                  },
                  regions:
polygonRegions,
                  regionColors:
colors),
              ),
            ],
          ),
        )
      ],
    ),
  ),
 );
 }
}

void
_modalBottomSheetInfografis(cont
ext, Function callback) {
 showModalBottomSheet(
  context: context,
  shape:
RoundedRectangleBorder(
   borderRadius:
BorderRadius.circular(20.0),
  ),
  backgroundColor: bgColor,
  builder: (BuildContext bc) {
   InfografisState infografisState =
bc.watch<InfografisBloc>().state;

   final infografisData =
infografisState.infografisList

.elementAt(infografisState.selected
List < 0
        ? 0
        :
infografisState.selectedList == 14
          ? 10
          :
infografisState.selectedList);

   final String language =
bc.locale.toString();

   return DefaultTabController(
    length: 3,
    initialIndex: 0,
    child: Column(
     children: <Widget>[
      TabBar(
       indicatorColor:
primaryColor,
       labelColor: primaryColor,
       unselectedLabelColor:
Colors.black87,
       labelStyle:
primaryStyle.copyWith(
        fontSize: 14,
        fontWeight:
FontWeight.bold,
       ),
       tabs: <Widget>[
        Tab(
         text:
"infografisModal.deskripsi".tr(),
        ),
        Tab(
         text:
"infografisModal.daging".tr(),
        ),
        Tab(
         text:
"infografisModal.resep".tr(),
        )
       ],
      ),
      Expanded(
       child: BlocProvider(
        create:    (context)    =>
ResepBloc()

..add(ResepLoaded(infografisState.
selectedList + 1)),
        child: TabBarView(
         children: [
          DeskripsiInfografis(
           infografisData:
infografisData,
           language: language,
          ),
          // _shimmerLoading(),
          ListDagingInfografis(),
          ListResepInfografis(),
         ],
        ),
       ),
      )
     ],
    ),
   );
  },
 ).whenComplete(() => callback());
}
```

mitosFakta.dart

```
import 'dart:ui';
import
'package:flutter/material.dart';
import
'package:easy_localization/easy_loc
alization.dart';
import
'package:flutter_bloc/flutter_bloc.d
art';
import
'package:flutter_spinkit/flutter_spin
kit.dart';
import
'package:infografis_daging_sapi/bl
oc/mitosfakta_bloc.dart';
import
'package:infografis_daging_sapi/sty
les/constant.dart';
import
'package:infografis_daging_sapi/wi
dgets/emptyData.dart';
import
'package:infografis_daging_sapi/wi
dgets/errorData.dart';

class MitosFaktaScreen extends
StatelessWidget {
 const MitosFaktaScreen({Key?
key}) : super(key: key);

 @override
 Widget build(BuildContext
context) {
  final String language =
context.locale.toString();

  return Scaffold(
    appBar: AppBar(
      title: Text(
        "menu.m&f",
        style:  TextStyle(fontWeight:
FontWeight.bold),
      ).tr(),
    ),
    backgroundColor: bgColor,
    body:
BlocBuilder<MitosFaktaBloc,
MitosFaktaState>(
      builder: (context, state) {
        return state.isLoading
            ? Column(
                mainAxisAlignment:
MainAxisAlignment.center,
                crossAxisAlignment:
CrossAxisAlignment.center,
                children: [
                  SpinKitFadingFour(
                    color: primaryColor,
                  ),
                  Text(
                    "Loading Data...",
                    style:
primaryStyle.copyWith(color:
Colors.black),
                  ),
                ],
              )
            : !state.error.status
                ? Row(
                    mainAxisAlignment:
MainAxisAlignment.center,
                    crossAxisAlignment:
CrossAxisAlignment.center,
                    children: [
                      errorData(),
                    ],
                  )
                :
state.mitosFaktaList.length <= 0
                    ? Row(

mainAxisAlignment:
MainAxisAlignment.center,

crossAxisAlignment:
CrossAxisAlignment.center,
                        children: [
                          emptyData(),
                        ],
                      )
                    : ListView.builder(
                        itemCount:
state.mitosFaktaList.length,
                        padding:
EdgeInsets.symmetric(vertical: 10),
                        itemBuilder:
(context, index) {
                          final mitosFakta =
state.mitosFaktaList[index];
                          return
IntrinsicHeight(
                            child: Row(

mainAxisAlignment:
MainAxisAlignment.start,
                              children: [
                                _cardMitos(

mitosFakta.mitos[language],
```

```
mitosFakta.mitos["imageUrl"],
                            ),
                            _cardFakta(

mitosFakta.fakta[language],

mitosFakta.fakta["imageUrl"],
                            )
                          ],
                        ),
                      );
                    });
        },
      ),
    );
  }
}

Widget _cardMitos(String judul,
String imageUrl) {
  return Expanded(
    child: Card(
      color: Color(0xff3b3838),
      child: Padding(
        padding:                const
EdgeInsets.all(8),
        child: Row(
          crossAxisAlignment:
CrossAxisAlignment.start,
          children: <Widget>[
            Expanded(
              child: Image.network(
                imageUrl,
                width: 10,
                fit: BoxFit.fill,
              ),
            ),
            Expanded(
              child: Column(
                crossAxisAlignment:
CrossAxisAlignment.end,
                children: [
                  Text(
                    "mitos",
                    style: TextStyle(
                      fontWeight:
FontWeight.bold,
                      fontSize: 14,
                      color: Colors.white,
                    ),
                  ).tr(),
                  Text(
                    judul,
                    textAlign:
TextAlign.end,
                    style:   TextStyle(color:
Colors.white),
                  ),
                ],
              ),
            ),
          ],
        ),
      ),
    ),
  );
}

Widget _cardFakta(String judul,
String imageUrl) {
  return Expanded(
    child: Card(
      color: Colors.white,
      child: Padding(
        padding:                const
EdgeInsets.all(8),
        child: Row(
          crossAxisAlignment:
CrossAxisAlignment.start,
          children: <Widget>[
            Expanded(
              child: Column(
                crossAxisAlignment:
CrossAxisAlignment.start,
                children: [
                  Text(
                    "fakta",
                    style: TextStyle(
                      fontWeight:
FontWeight.bold,
                      fontSize: 14,
                      color:
Color(0xFF2b2b2b),
                    ),
                  ).tr(),
                  Text(
                    judul,
                  ),
                ],
              ),
            ),
            Expanded(
              child: Image.network(
                imageUrl,
                width: 10,
                fit: BoxFit.fill,
              ),
            ),
```

```
            ],
          ),
        ),
      ),
    );
}
```

anekaOlahan.dart

```
import 'package:flutter/material.dart';
import 'package:easy_localization/easy_localization.dart';
import 'package:flutter_bloc/flutter_bloc.dart';
import 'package:flutter_spinkit/flutter_spinkit.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:infografis_daging_sapi/bloc/olahan_bloc.dart';
import 'package:infografis_daging_sapi/models/olahan_models.dart';
import 'package:infografis_daging_sapi/models/resep_models.dart';
import 'package:infografis_daging_sapi/styles/constant.dart';
import 'package:infografis_daging_sapi/widgets/emptyData.dart';
import 'package:infografis_daging_sapi/widgets/errorData.dart';

import 'resep.dart';

class AnekaOlahanScreen extends StatelessWidget {
  const AnekaOlahanScreen({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    TextEditingController controller = TextEditingController();
    final String language = context.locale.toString();
    List<OlahanModel> olahanList = context.select<OlahanBloc, List<OlahanModel>>(
          (OlahanBloc olahanBloc) => olahanBloc.state.olahanList);

    String selectedMetode = context.select<OlahanBloc, String>(
        (OlahanBloc olahanBloc) => olahanBloc.state.selectedMetode);

    return Scaffold(
      appBar: AppBar(
        title: Text(
          "menu.olahan",
          style: TextStyle(fontWeight: FontWeight.bold),
        ).tr(),
      ),
      backgroundColor: bgColor,
      body: Column(
        children: <Widget>[
          new Container(
            child: new Padding(
              padding: const EdgeInsets.all(10),
              child: new Card(
                shape: RoundedRectangleBorder(
                  borderRadius: BorderRadius.circular(20.0),
                ),
                child: new ListTile(
                  leading: new Icon(Icons.search),
                  title: new TextField(
                    controller: controller,
                    decoration: new InputDecoration(
                      hintText: 'cari'.tr(),
                      border: InputBorder.none,
                    ),
                    onChanged: (value) {

context.read<OlahanBloc>().add(FilterBySearch(
                          olahanList: olahanList,
                          query: value,
                          language: language,
                        ));
                    },
```

```
                ),
                trailing:              new
IconButton(
                  icon:                new
Icon(Icons.cancel),
                  onPressed: () {
                    controller.clear();
                    context

.read<OlahanBloc>()

.add(ClearFilter(olahanList:
olahanList));
                  },
                ),
              ),
            ),
          ),
        ),
        Row(
          mainAxisAlignment:
MainAxisAlignment.center,
          crossAxisAlignment:
CrossAxisAlignment.center,
          children: <Widget>[
            _buttonMetode(context,
"bbq", selectedMetode, olahanList),
            _buttonMetode(context,
"rebus",              selectedMetode,
olahanList),
            _buttonMetode(context,
"goreng",             selectedMetode,
olahanList),
            _buttonMetode(context,
"panggang",           selectedMetode,
olahanList),
            _buttonMetode(context,
"oven",               selectedMetode,
olahanList),
          ],
        ),
        Expanded(
          child: Container(
            width:
MediaQuery.of(context).size.width,
            color: Colors.white,
            child:
BlocBuilder<OlahanBloc,
OlahanState>(
              builder: (context, state)
{
                return state.isLoading
                  ? Column(
                      mainAxisAlignment:
MainAxisAlignment.center,

crossAxisAlignment:
CrossAxisAlignment.center,
                      children: [

SpinKitFadingFour(
                          color:
primaryColor,
                        ),
                        Text(
                          "Loading
Data...",
                          style:

primaryStyle.copyWith(color:
Colors.black),
                        ),
                      ],
                    )
                  : !state.error.status
                    ? errorData()
                    :
state.olahanListFilter.length <= 0
                      ? emptyData()
                      :
ListView.builder(
                          shrinkWrap:
true,
                          itemCount:
state.olahanListFilter.length,
                          padding:
EdgeInsets.all(10),
                          itemBuilder:
(context, index) {
                            final olahan
=

state.olahanListFilter[index];
                            return
Container(
                              margin:
EdgeInsets.symmetric(
                                vertical:
4,
                              ),
                              height: 60,
                              width:
double.infinity,

decoration: BoxDecoration(
                                color:
Colors.white,
```

```
borderRadius: BorderRadius.only(
topRight: Radius.circular(25),
bottomRight: Radius.circular(25)),
                                ),
                                child:
Material(
                                  color:
bgColor,
borderRadius:
BorderRadius.circular(25),
                                elevation:
3,
shadowColor: Colors.white,
                                child:
InkWell(
borderRadius:
BorderRadius.circular(25),
                                  onTap:
() {
                                    final
route = MaterialPageRoute(
builder: (context) =>
ResepScreen(
resepData:
ResepModel.fromJson(
olahan.toJson()),
                                      ),
                                    );
Navigator.push(context, route);
                                  },
                                child:
ListTile(
leading: CircleAvatar(
radius: 25,
backgroundColor:
Colors.transparent,
backgroundImage: NetworkImage(
olahan.imageUrl),
                                      ),
                                      title:
Text(
olahan.nama[language],
                                      ),
trailing: new Icon(
Icons.arrow_right_alt,
color: Colors.black,
                                      )),
                                  ),
                                ),
                              );
                            });
                      },
                    ),
                  ),
                ),
              ],
            ));
  }
}

Widget _buttonMetode(
  BuildContext context,
  String metode,
  String selectedMetode,
  List<OlahanModel> olahanList,
) {
  return IconButton(
    icon: Image.asset(
"assets/images/${metode}_icon${s
electedMetode == metode ?
'_active' : ''}.png"),
    iconSize: 50,
    onPressed: () {
      if (selectedMetode == metode)
context.read<OlahanBloc>().add(Cl
earFilter(olahanList: olahanList));
      else
context.read<OlahanBloc>().add(
            FilterByMetode(
              olahanList: olahanList,
              metode: metode,
            ),
```

```
            );

        Fluttertoast.showToast(
          msg: metode.toUpperCase(),
          toastLength:
Toast.LENGTH_SHORT,
          gravity:
ToastGravity.CENTER,
          backgroundColor: bgColor,
          textColor: Colors.black87,
          fontSize: 16.0,
        );
      },
    );
}
```

recep.dart

```
import
'package:flutter/material.dart';
import
'package:easy_localization/easy_loc
alization.dart';
import
'package:infografis_daging_sapi/m
odels/resep_models.dart';
import
'package:infografis_daging_sapi/ser
vices/tts_service.dart';
import
'package:infografis_daging_sapi/sty
les/constant.dart';

class ResepScreen extends
StatelessWidget {
  const ResepScreen({
    Key? key,
    required this.resepData,
  }) : super(key: key);

  final ResepModel resepData;

  @override
  Widget build(BuildContext
context) {
    TtsService ttsService =
        TtsService(resepData:
resepData, language:
context.locale.toString());
    final String language =
context.locale.toString();

    return Scaffold(
      backgroundColor: Colors.white,
      body: CustomScrollView(
        slivers: <Widget>[
          SliverAppBar(
            pinned: true,
            iconTheme:
IconThemeData(color:
dangerColor),
            expandedHeight: 300,
            backgroundColor: bgColor,
            flexibleSpace:
FlexibleSpaceBar(
              title: Text(

resepData.nama[language],
                style: TextStyle(
                  color: Colors.white,
                  shadows: <Shadow>[
                    Shadow(
                      offset:
Offset(1.0,
1.0),
                      blurRadius: 3.0,
                      color:
Color.fromARGB(255, 0, 0, 0),
                    ),
                  ],
                ),
              ),
              background:
Image.network(
                resepData.imageUrl,
                // fit: BoxFit.cover,
              ),
            ),
          ),
          SliverPadding(
            padding: EdgeInsets.all(20),
            sliver: SliverList(
              delegate:
SliverChildListDelegate([
                Row(
                  children: <Widget>[
                    Expanded(
                      child: Stack(
                        clipBehavior:
Clip.none,
                        alignment:
AlignmentDirectional.bottomStart,
                        children: <Widget>[
                          SizedBox(height:
40),
                          Positioned(
                            child: Text(
                              "resep.bahan",
                              style:
secondaryStyle,
                            ).tr(),
                          )
```

```
                    ],
                  ),
                ),
                ElevatedButton(
                  onPressed: () =>

_modalBottomSheetTts(context,
resepData, ttsService),
                  child: Icon(

Icons.record_voice_over,
                    color: Colors.white,
                  ),
                  style:
ElevatedButton.styleFrom(
                    onPrimary: bgColor,
                    primary:
primaryColor,
                    minimumSize:
Size(50, 36),
                    padding:
EdgeInsets.symmetric(horizontal:
10),
                    shape: const
RoundedRectangleBorder(
                      borderRadius:
BorderRadius.all(Radius.circular(1
0)),
                    ),
                  ),
                )
              ],
            ),
            Divider(),
            for (var i = 0; i <
resepData.bahan[language].length;
i++)
              Padding(
                padding: const
EdgeInsets.only(bottom: 5),
                child: new
Text(resepData.bahan[language][i])
,
              ),
            Divider(),
            Text(
              "resep.cara",
              style: secondaryStyle,
            ).tr(),
            Divider(),
            for (var i = 0; i <
resepData.cara[language].length;
i++)
              Padding(
                padding: const
EdgeInsets.only(bottom: 5),
                child: Row(
                  mainAxisAlignment:
MainAxisAlignment.start,
                  crossAxisAlignment:
CrossAxisAlignment.start,
                  children: <Widget>[
                    Text("${i + 1}. "),
                    Expanded(
                      child: new Text(

resepData.cara[language][i],
                        textAlign:
TextAlign.justify,
                      )),
                  ],
                ),
              ),
            Divider(),
            Text(
              "resep.porsi",
              style: secondaryStyle,
            ).tr(args:
[resepData.porsi.toString()]),
            Divider(),
            Container(
              padding:
EdgeInsets.symmetric(
                horizontal: 25,
                vertical: 10,
              ),
              decoration:
BoxDecoration(
                color: primaryColor,
                borderRadius:
BorderRadius.circular(10),
              ),
              child: Column(
                crossAxisAlignment:
CrossAxisAlignment.start,
                children: [
                  Text(
                    "resep.gizi",
                    style:

primaryStyle.copyWith(fontWeight
: FontWeight.bold),
                  ).tr(),
                  Text(
                    "resep.energi",
                    style: primaryStyle,
                  ).tr(args:
[resepData.gizi['energi']]),
                  Text(
```

```
                    "resep.kolestrol",
                    style: primaryStyle,
                  ).tr(args:
[resepData.gizi['kolestrol']]),
                ],
              ),
            )
          ]),
        ),
      ),
    ],
  ),
 );
 }
}

void _modalBottomSheetTts(
 context,
 ResepModel resepData,
 TtsService ttsService,
) {
 showModalBottomSheet(
   context: context,
   shape:
RoundedRectangleBorder(
     borderRadius:
BorderRadius.circular(20.0),
   ),
   backgroundColor: bgColor,
   builder: (BuildContext bc) {
    return Container(
     height: 100,
     color: bgColor,
     child: Center(
      child: Row(
       mainAxisAlignment:
MainAxisAlignment.center,
       crossAxisAlignment:
CrossAxisAlignment.center,
       children: <Widget>[
        ElevatedButton(
         child: Padding(
          padding:          const
EdgeInsets.all(8.0),
          child: Row(
           mainAxisAlignment:
MainAxisAlignment.center,
           crossAxisAlignment:
CrossAxisAlignment.center,
           mainAxisSize:
MainAxisSize.min,
           children: [
            const
Icon(Icons.play_arrow),
            Text("mulai").tr(),
           ],
          ),
         ),
         onPressed:    ()    =>
ttsService.speak(),
        ),
        SizedBox(width: 10),
        ElevatedButton(
         child: Padding(
          padding:          const
EdgeInsets.all(8.0),
          child: Row(
           mainAxisAlignment:
MainAxisAlignment.center,
           crossAxisAlignment:
CrossAxisAlignment.center,
           mainAxisSize:
MainAxisSize.min,
           children: [
            const
Icon(Icons.stop),
            Text("berhenti").tr(),
           ],
          ),
         ),
         onPressed:    ()    =>
ttsService.stop(),
        ),
       ],
      ),
     ),
    );
   });
}
```

tentang.dart

```
import
'package:flutter/material.dart';
import
'package:easy_localization/easy_loc
alization.dart';
import
'package:fluttertoast/fluttertoast.dar
t';
import
'package:infografis_daging_sapi/sty
les/constant.dart';

class TentangScreen extends
StatelessWidget {
 const TentangScreen({Key? key})
: super(key: key);

 @override
```

```
 Widget build(BuildContext context) {
   return Scaffold(
     appBar: AppBar(
       title: Text(
         "menu.tentang",
         style:  TextStyle(fontWeight: FontWeight.bold),
       ).tr(),
     ),
     backgroundColor: bgColor,
     body: Card(
       margin: EdgeInsets.symmetric(horizontal: 20, vertical: 20),
       shape: RoundedRectangleBorder(
         borderRadius: BorderRadius.circular(20.0),
       ),
       child: Column(
         crossAxisAlignment: CrossAxisAlignment.start,
         children: [
           SizedBox(height: 20),
           Center(
             child: Image(
               image: AssetImage("assets/images/logo.png"),
               width: 150,
             ),
           ),
           ListTile(
             title: const Text('tentang.judul').tr(),
           ),
           Padding(
             padding: const EdgeInsets.symmetric(horizontal: 16),
             child: Text(
               'tentang.info',
               style: TextStyle(
                 color: Colors.black.withOpacity(0.6),
               ),
               textAlign: TextAlign.justify,
             ).tr(),
           ),
           Padding(
             padding: const EdgeInsets.all(16.0),
             child: Center(
               child: Text(
                 'Syaiful \n217280165',
                 style: TextStyle(fontSize: 20),
                 textAlign: TextAlign.center,
               ),
             ),
           ),
           Expanded(
             child: Padding(
               padding: const EdgeInsets.symmetric(horizontal: 16),
               child: Column(
                 crossAxisAlignment: CrossAxisAlignment.start,
                 children: <Widget>[

                   _dosenInfo("pembimbing", "1", "Muhammad Zainal, ST. MT"),
                   SizedBox(height: 10),

                   _dosenInfo("pembimbing", "2", "Masnur, ST., M.Kom"),
                   SizedBox(height: 10),
                   _dosenInfo("penguji", "1", "Marlina, S.kom, M.Kom"),
                   SizedBox(height: 10),
                   _dosenInfo("penguji", "2", "Hj. A. Irmayani P, ST., MT"),
                 ],
               ),
             ),
           ),
           Row(
             crossAxisAlignment: CrossAxisAlignment.center,
             mainAxisAlignment: MainAxisAlignment.center,
             children: <Widget>[
               InkWell(
                 child: Container(
                   child: Image.asset(

                     "assets/images/google_play.png",
                     height: 25,
                   ),
                 ),
                 onTap: () {

                   Fluttertoast.showToast(
                     msg: "Coming Soon",
                     toastLength: Toast.LENGTH_SHORT,
```

```
                gravity:
ToastGravity.SNACKBAR,
                backgroundColor:
bgColor,
                textColor:
Colors.black87,
                fontSize: 16.0,
               );
             },
           ),
           SizedBox(width: 10),
           InkWell(
             child: Container(
               child: Image.asset(

"assets/images/gmail.png",
                 height: 30,
               ),
             ),
             onTap: () {

Fluttertoast.showToast(
                 msg:
"syaifulsaipul99@gmail.com",
                 toastLength:
Toast.LENGTH_SHORT,
                 gravity:
ToastGravity.BOTTOM,
                 backgroundColor:
bgColor,
                 textColor:
Colors.black87,
                 fontSize: 16.0,
               );
             },
           ),
         ],
       ),
       SizedBox(height: 20)
     ],
   ),
  ),
 );
 }
}

Widget _dosenInfo(String dosen,
String angka, String nama) {
 final dosenLength =
"tentang.$dosen".tr().length;
 return Row(
  children: [
   Text("tentang.$dosen").tr(args:
[angka]),
   SizedBox(
     width: dosen ==
"pembimbing"
         ? 20
         : dosenLength == 11
           ? 25
           : 52),
   Text(":"),
   SizedBox(width: 20),
   Expanded(child: Text(nama)),
  ],
 );
}
```

main.dart

```
import
'package:easy_localization/easy_loc
alization.dart';
import
'package:flutter/material.dart';
import
'package:flutter/services.dart';
import
'package:flutter_bloc/flutter_bloc.d
art';
import 'bloc/infografis_bloc.dart';
import 'bloc/olahan_bloc.dart';
import 'bloc/mitosfakta_bloc.dart';
import 'styles/constant.dart';
import 'screens/home.dart';
import 'screens/infografis.dart';
import 'screens/mitosFakta.dart';
import 'screens/anekaOlahan.dart';
import 'screens/tentang.dart';

void main() async {

WidgetsFlutterBinding.ensureInitial
ized();
 await
EasyLocalization.ensureInitialized(
);
 runApp(
  EasyLocalization(
   supportedLocales:
[Locale("id"), Locale("en")],
   path: 'assets/translations',
   fallbackLocale: Locale("id"),
   child: MyApp(),
  ),
 );
}

class MyApp extends
StatelessWidget {
```

```
 const MyApp({Key? key}) : super(key: key);

 @override
 Widget build(BuildContext context) {

SystemChrome.setSystemUIOverlayStyle(

SystemUiOverlayStyle.dark.copyWith(
        statusBarColor: Colors.transparent,
        statusBarIconBrightness: Brightness.light,
        statusBarBrightness: Brightness.light),
   );

   return MultiBlocProvider(
     providers: [
       BlocProvider(
         create: (context) => InfografisBloc(),
       ),
       BlocProvider(
         create: (context) => OlahanBloc()..add(OlahanLoaded()),
       ),
       BlocProvider(
         create: (context) => MitosFaktaBloc()..add(MitosFaktaLoaded()),
       ),
     ],
     child: MaterialApp(
       // debugShowCheckedModeBanner: false,
       localizationsDelegates: context.localizationDelegates,
       supportedLocales: context.supportedLocales,
       locale: context.locale,
       theme: ThemeData(
         // scaffoldBackgroundColor: bgColor,
         primaryColor: primaryColor,
         splashColor: secondaryColor,
         primarySwatch: Colors.red,
         fontFamily: 'Gothic',
         appBarTheme: Theme.of(context)
             .appBarTheme
             .copyWith(brightness: Brightness.dark),
       ),
       routes: {
         '/': (context) => HomeScreen(),
         '/infografis': (context) => InfografisScreen(),
         '/mitosFakta': (context) => MitosFaktaScreen(),
         '/anekaOlahan': (context) => AnekaOlahanScreen(),
         '/tentang': (context) => TentangScreen(),
       },
     ),
   );
 }
}
```

generated.dart

```
//
// Generated file. Do not edit.
//

// ignore_for_file: lines_longer_than_80_chars

import 'package:flutter_tts/flutter_tts_web.dart';
import 'package:fluttertoast/fluttertoast_web.dart';
import 'package:shared_preferences_web/shared_preferences_web.dart';

import 'package:flutter_web_plugins/flutter_web_plugins.dart';

// ignore: public_member_api_docs
void registerPlugins(Registrar registrar) {

FlutterTtsPlugin.registerWith(registrar);

FluttertoastWebPlugin.registerWith(registrar);

SharedPreferencesPlugin.registerWith(registrar);
```

```
    registrar.registerMessageHandler();
}
```

menuHome.dart

```
import
'package:flutter/material.dart';
import '../styles/constant.dart';

class MenuWidget extends
StatelessWidget {
  MenuWidget({required this.text,
required this.url, required
this.img});

  final String text;
  final String url;
  final String img;

  @override
  Widget build(BuildContext
context) {
    return Container(
      margin:
EdgeInsets.symmetric(horizontal:
20),
      height: 60,
      width: double.infinity,
      decoration: BoxDecoration(
        borderRadius:
BorderRadius.circular(25),
      ),
      child: Material(
        color: bgColor,
        borderRadius:
BorderRadius.circular(25),
        elevation: 3,
        shadowColor: Colors.white,
        child: InkWell(
          onTap: () {

Navigator.pushNamed(context,
this.url);
          },
          borderRadius:
BorderRadius.circular(25),
          child: Padding(
            padding:
EdgeInsets.only(left: 20),
            child:
Row(children:
<Widget>[
              Image(
                image:
AssetImage(this.img),
                width: 60,
                height: 50,
              ),
              SizedBox(width: 25),
              Expanded(
                child: Text(
                  this.text,
                  style:
primaryStyle.copyWith(
                    fontSize: 20,
                    color: Colors.black,
                  ),
                ),
              ),
              Container(
                decoration:
BoxDecoration(
                  color: Colors.white,
                  borderRadius:
BorderRadius.only(
                    topRight:
Radius.circular(25),
                    bottomRight:
Radius.circular(25)),
                ),
                height: double.infinity,
                width: 50,
                child: Icon(

Icons.arrow_right_alt_rounded,
                  color: primaryColor,
                ),
              )
            ]),
          ),
        ),
      ),
    );
```

loading.dart

```
import
'package:flutter/material.dart';
import
'package:flutter_spinkit/flutter_spin
kit.dart';
import
'package:infografis_daging_sapi/sty
les/constant.dart';

class Loading extends
ModalRoute<void> {
  @override
  Duration get transitionDuration
=> Duration(milliseconds: 500);

  @override
```

```
  bool get opaque => false;

  @override
  bool get barrierDismissible => false;

  @override
  Color get barrierColor => Colors.black.withOpacity(0.5);

  @override
  String? get barrierLabel => null;

  @override
  bool get maintainState => true;

  @override
  Widget buildPage(
    BuildContext context,
    Animation<double> animation,
    Animation<double> secondaryAnimation,
  ) {
    return Material(
      type: MaterialType.transparency,
      child: SafeArea(
        child: _buildOverlayContent(context),
      ),
    );
  }

  Widget _buildOverlayContent(BuildContext context) {
    return Column(
      mainAxisAlignment: MainAxisAlignment.center,
      crossAxisAlignment: CrossAxisAlignment.center,
      children: [
        SpinKitFadingFour(
          color: bgColor,
        ),
        Text(
          "Loading Data..",
          style: primaryStyle,
        ),
      ],
    );
  }

  @override
  Widget buildTransitions(BuildContext context, Animation<double> animation,
      Animation<double> secondaryAnimation, Widget child) {
    // You can add your own animations for the overlay content
    return FadeTransition(
      opacity: animation,
      child: ScaleTransition(
        scale: animation,
        child: child,
      ),
    );
  }
}
```

listRecepInfografis.dart

```
import 'package:flutter/material.dart';
import 'package:easy_localization/easy_localization.dart';
import 'package:flutter_bloc/flutter_bloc.dart';
import 'package:infografis_daging_sapi/bloc/resep_bloc.dart';
import 'package:infografis_daging_sapi/screens/resep.dart';
import 'package:infografis_daging_sapi/styles/constant.dart';
import 'package:shimmer/shimmer.dart';

import 'emptyData.dart';

class ListRecepInfografis extends StatelessWidget {
  const ListRecepInfografis({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    final String language = context.locale.toString();

    return BlocBuilder<ResepBloc, ResepState>(
```

```
    builder: (context, state) {
      return state.isLoading
          ? _shimmerLoading()
          : state.resepList.length <= 0
              ? emptyData()
              : ListView.builder(
                  itemCount:
state.isLoading ? 5 :
state.resepList.length,
                  padding:
EdgeInsets.symmetric(vertical: 10),
                  itemBuilder: (context,
index) {
                    final resepList =
state.resepList[index];

                    return Container(
                      margin:
EdgeInsets.symmetric(
                        horizontal: 15,
                        vertical: 3,
                      ),
                      height: 60,
                      width:
double.infinity,
                      decoration:
BoxDecoration(
                        borderRadius:
BorderRadius.circular(20),
                      ),
                      child: Material(
                        color:
Colors.white,
                        borderRadius:
BorderRadius.circular(20),
                        elevation: 3,
                        shadowColor:
Colors.white,
                        child: InkWell(
                          onTap: () {
                            final route =
MaterialPageRoute(
                              builder:
(context) => ResepScreen(
                                resepData:
resepList,
                              ),
                            );

Navigator.push(context, route);
                          },
                          borderRadius:
BorderRadius.circular(20),
                          child: Padding(
                            padding:
EdgeInsets.only(left: 20),
                            child:
Row(children: <Widget>[
                              Image.network(

resepList.imageUrl,
                                height: 50,
                              ),

SizedBox(width: 25),
                              Expanded(
                                child: Text(

resepList.nama[language],
                                  style:
primaryStyle.copyWith(
                                    fontSize:
18,
                                    color:
Colors.black,
                                  ),
                                ),
                              ),
                              Container(
                                decoration:
BoxDecoration(
                                  color:
primaryColor,

borderRadius: BorderRadius.only(
                                    topRight:
Radius.circular(20),

bottomRight: Radius.circular(20),
                                  ),
                                ),
                                height:
double.infinity,
                                width: 50,
                                child: Icon(

Icons.arrow_right_alt_rounded,
                                  color:
Colors.white,
                                ),
                              )
                            ]),
                          ),
                        ),
                      ),
                    );
                  },
                );
    },
```

```
    );
  }
}

Widget _shimmerLoading() {
  return Shimmer.fromColors(
    baseColor: Colors.white,
    highlightColor: bgColor,
    child: ListView.builder(
      itemCount: 7,
      itemBuilder: (context, index) {
        return Container(
          margin:
EdgeInsets.symmetric(
            horizontal: 15,
            vertical: 3,
          ),
          height: 60,
          width: double.infinity,
          decoration: BoxDecoration(
            borderRadius:
BorderRadius.circular(20),
          ),
          child: Material(
            color: Colors.white,
            borderRadius:
BorderRadius.circular(20),
            elevation: 3,
            shadowColor: Colors.white,
          ),
        );
      },
    ),
  );
}
```

listDagingInfografis.dart

```
import
'package:flutter/material.dart';
import
'package:easy_localization/easy_loc
alization.dart';
import
'package:flutter_bloc/flutter_bloc.d
art';
import
'package:infografis_daging_sapi/bl
oc/infografis_bloc.dart';
import
'package:infografis_daging_sapi/sty
les/constant.dart';

import 'emptyData.dart';

class ListDagingInfografis extends
StatelessWidget {
  const ListDagingInfografis({Key?
key}) : super(key: key);

  @override
  Widget build(BuildContext
context) {
    final String language =
context.locale.toString();

    return
BlocBuilder<InfografisBloc,
InfografisState>(
      builder: (context, state) {
        return
state.dagingListFilter.length <= 0
            ? emptyData()
            : ListView.builder(
                itemCount:
state.dagingListFilter.length,
                padding:
EdgeInsets.symmetric(vertical: 10),
                itemBuilder:       (context,
index) {
                  final dagingList =
state.dagingListFilter[index];
                  return Card(
                    margin:
EdgeInsets.symmetric(
                      vertical: 3,
                      horizontal: 15,
                    ),
                    shape:
RoundedRectangleBorder(
                      borderRadius:
BorderRadius.circular(20.0),
                    ),
                    child: Material(
                      color: Colors.white,
                      borderRadius:
BorderRadius.circular(20),
                      child: InkWell(
                        onTap:       ()       =>
print(index),
                        borderRadius:
BorderRadius.circular(20),
                        splashColor:
bgColor,
                        child: ListTile(
                          leading:
CircleAvatar(
                            radius: 25,
                            backgroundColor:
Colors.transparent,
```

```
                backgroundImage:
NetworkImage(dagingList.imageUr
l),
                          ),
                          title:
Text(dagingList.nama[language]
                              .toString()

.capitalizeFirstofEach),
                          subtitle:
Text(dagingList.deskripsi[language
]),
                        ),
                      ),
                    ),
                  );
                },
              );
            },
          );
        }
      }

      extension CapExtension on String {
        String get inCaps =>
            this.length > 0 ?
'${this[0].toUpperCase()}${this.sub
string(1)}' : '';
        String get allInCaps =>
this.toUpperCase();
        String get capitalizeFirstofEach
=> this
            .replaceAll(RegExp(' +'), ' ')
            .replaceAll('.', '')
            .split(" ")
            .map((str) => str.inCaps)
            .join(" ");
}
```

headerHome.dart

```
import
'package:flutter/material.dart';
import
'package:easy_localization/easy_loc
alization.dart';

class HeaderHome extends
StatelessWidget {
  const HeaderHome({Key? key}) :
super(key: key);

  @override
  Widget build(BuildContext
context) {
    return ClipPath(
      clipper: MyClipper(),
      child: Container(
        height: 290,
        width: double.infinity,
        decoration: BoxDecoration(
          gradient: LinearGradient(
            begin:
Alignment.bottomLeft,
            end: Alignment.topRight,
            colors: [
              Color(0xFFfa6666),
              Color(0xFFfa0a0a),
              Color(0xFFfa6666),
            ]),
        ),
        child: SafeArea(
          child: Column(
            crossAxisAlignment:
CrossAxisAlignment.start,
            children: <Widget>[
              SizedBox(height: 20),
              Expanded(
                child: Stack(
                children: <Widget>[
                  Transform.scale(
                    scale: 0.9,
                    child: Image(
                      image:
AssetImage("assets/images/cow.pn
g"),
                      width: 190,
                    ),
                  ),
                  Positioned(
                    top: 30,
                    left: 195,
                    child: Text(
                      "app_title",
                      style: TextStyle(
                        fontSize: 30,
                        color: Colors.white,
                        shadows:
<Shadow>[
                          Shadow(
                            offset:  Offset(3.0,
3.0),
                            blurRadius: 3.0,
                            color:
Color.fromARGB(255, 0, 0, 0),
                          ),
                        ],
                      ),
                    ).tr(),
                  ),
```

```
                Container(),
              ],
            ))
          ],
        ),
      ),
    ),
  );
 }
}

class MyClipper extends
CustomClipper<Path> {
 @override
 Path getClip(Size size) {
  var path = Path();
  path.lineTo(0, size.height - 80);
  path.quadraticBezierTo(
   size.width / 2,
   size.height,
   size.width,
   size.height - 80,
  );
  path.lineTo(size.width, 0);
  path.close();
  return path;
 }

 @override
 bool shouldReclip(covariant
CustomClipper<Path> oldClipper)
{
  return false;
 }
}
```

errorData.dart

```
import
'package:flutter/material.dart';
import
'package:easy_localization/easy_loc
alization.dart';
import
'package:infografis_daging_sapi/sty
les/constant.dart';

Widget errorData() {
 return Column(
  crossAxisAlignment:
CrossAxisAlignment.center,
  mainAxisAlignment:
MainAxisAlignment.center,
  children: <Widget>[
   Icon(
    Icons.signal_wifi_connected_no_in
ternet_4,
    size: 50,
    color: secondaryColor,
   ),
   Text(
    "error_data",
    style:
secondaryStyle.copyWith(color:
Colors.black54),
   ).tr(),
  ],
 );
}
```

emptyData.dart

```
import
'package:flutter/material.dart';
import
'package:easy_localization/easy_loc
alization.dart';
import
'package:infografis_daging_sapi/sty
les/constant.dart';

Widget emptyData() {
 return Column(
  crossAxisAlignment:
CrossAxisAlignment.center,
  mainAxisAlignment:
MainAxisAlignment.center,
  children: <Widget>[
   Icon(
    Icons.no_food,
    size: 50,
    color: secondaryColor,
   ),
   Text(
    "empty_data",
    style:
secondaryStyle.copyWith(color:
Colors.black54),
   ).tr(),
  ],
 );
}
```

deskripsiInfografis.dart

```
import
'package:flutter/material.dart';
import
'package:infografis_daging_sapi/m
odels/infografis_models.dart';
```

```
class DeskripsiInfografis extends StatelessWidget {
  const DeskripsiInfografis({
    Key? key,
    required this.infografisData,
    required this.language,
  }) : super(key: key);

  final InfografisModel infografisData;
  final String language;

  @override
  Widget build(BuildContext context) {
    return ListView(
      padding: EdgeInsets.symmetric(horizontal: 20, vertical: 10),
      children: <Widget>[
        Center(
          child: Text(

infografisData.nama[language],
            style: TextStyle(fontSize: 20),
          ),
        ),
        Divider(),

Text(infografisData.deskripsi[language]["ket"]),
        Divider(),
        SizedBox(height: 10),
        _contentDeskripsi(
          "Warna",

infografisData.deskripsi[language]["warna"],
        ),
        SizedBox(height: 10),
        _contentDeskripsi(
          "Aroma",

infografisData.deskripsi[language]["aroma"],
        ),
        SizedBox(height: 10),
        _contentDeskripsi(
          "Penampakan",

infografisData.deskripsi[language]["penampakan"],
        ),
        SizedBox(height: 10),
        _contentDeskripsi(
          "Kekenyalan",

infografisData.deskripsi[language]["kekenyalan"],
        ),
      ],
    );
  }
}

Widget _contentDeskripsi(String title, String content) {
  return Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: [
      Row(
        children: [
          Expanded(child: Divider()),
          SizedBox(width: 5),
          Text(title),
          SizedBox(width: 5),
          Expanded(child: Divider()),
        ],
      ),
      Text(content),
      Divider()
    ],
  );
}
```

alertError.dart

```
import 'package:flutter/material.dart';
import 'package:rflutter_alert/rflutter_alert.dart';
import 'package:easy_localization/easy_localization.dart';

import 'errorData.dart';

Future alertError(BuildContext context) {
  return Alert(
    context: context,
    title: "",
    content: errorData(),
    style: AlertStyle(
      animationType: AnimationType.fromBottom,
      isCloseButton: false,
      isOverlayTapDismiss: false,
```

```
        descStyle:
    TextStyle(fontWeight:
    FontWeight.bold),
        descTextAlign: TextAlign.start,
        animationDuration:
    Duration(milliseconds: 400),
        alertBorder:
    RoundedRectangleBorder(
          borderRadius:
    BorderRadius.circular(0.0),
          side: BorderSide(
            color: Colors.grey,
          ),
        ),
      ),
      buttons: [
        DialogButton(
          child: Text(
            "tutup".tr(),
            style:          TextStyle(color:
    Colors.white, fontSize: 20),
          ),
          onPressed:          ()          =>
    Navigator.pop(context),
          radius:
    BorderRadius.circular(0.0),
        ),
      ],
    ).show();
}
```